Optimization Based Self-Localization for IoT Wireless Sensor Networks

Paul Beuchat, Henrik Hesse, Alexander Domahidi, and John Lygeros

Abstract—**In this paper we propose an embedded optimization framework for the simultaneous self-localization of all sensors in wireless sensor networks making use of range measurements from ultra-wideband (UWB) signals. Low-power UWB radios, which provide time-of-arrival measurements with decimeter accuracy over large distances, have been increasingly envisioned for real-time localization of IoT devices in GPS-denied environments and large sensor networks. In this work, we therefore explore different non-linear least-squares optimization problems to formulate the localization task based on UWB range measurements. We solve the resulting optimization problems directly using non-linear-programming algorithms that guarantee convergence to locally optimal solutions. This optimization framework allows the consistent comparison of different optimization methods for sensor localization. We propose and demonstrate the best optimization approach for the self-localization of sensors equipped with off-the-shelf microcontrollers using state-of-the-art code generation techniques for the plug-and-play deployment of the optimal localization algorithm. Numerical results indicate that the proposed approach improves localization accuracy and decreases computation times relative to existing iterative methods.**

Index Terms—**Localization, Ultra-Wideband Ranging, Non-Linear Embedded Optimization, Wireless Sensor Networks**

I. Introduction

FUTURE applications of IoT sensor networks, such as autonomous warehousing, safe mining, or autonomous driving, increasingly require the simultaneous localization of a large number of sensors, sometimes up to centimeter accuracy, in indoor or GPS-denied environments. In this context, ultra-wideband (UWB) communication has been explored as a low-cost, low-power means to localize sensors [1]. In this work we explore embedded optimization techniques to self-localize IoT devices based on the time of arrival (ToA), or transmission delay, of UWB signals measured at the receiving sensor. The latest technology of UWB radios can produce distance estimates from ToA measurements with decimeter accuracy over a range of hundred meters [2]. Such accuracy in addition to high sampling rate, ability to penetrate obstacles, and very low cost, makes UWB-based ranging a very appealing technology for localization of sensors in cluttered [3], dynamic [4], [5] or indoor [6], [7] environments where GPS-based navigation would fail. We refer the reader to [8] for an overview of the UWB technology and [9] for applications in IoT scenarios.

P. Beuchat and J. Lygeros are with the Automatic Control Laboratory at ETH Zurich, Physikstrasse 3, 8092 Zurich, Switzerland. E-mail: {beuchatp,jlygeros}@ethz.ch

H. Hesse is with the Aerospace Sciences Division, University of Glasgow (Singapore), 510 Dover Road, Singapore 139660. E-mail: henrik.hesse@glasgow.ac.uk.

A. Domahidi is with embotech AG, Physikstrasse 3, Zurich 8092, Switzerland. E-mail: domahidi@embotech.com.

This paper focuses on embedded optimization methods for localization using current off-the-shelf IoT microcontrollers with limited processing power. We use the FORCES Pro optimization environment [10], [11] to generate efficient solver code that implements iterative non-linear programming (NLP) algorithms for solving constrained non-linear optimization problems. Despite the limitations in processing power in IoT devices, the generated code can be ported onto off-the-shelf IoT microcontrollers in a plug-and-play fashion to enable the use of optimization methods for embedded sensor localization. This is key to developing a localization system that is easy to deploy, self-contained, and robust to changes in the arrangement without needing additional compute power.

Optimization approaches that directly consider the localization as a least squares problem have been investigated in the literature [12], [13]. This includes recent work that solves a sequence of convex optimization problems and converges to an improved estimate [14], referred to as multi-dimensional scaling (MDS). Moreover, non-linear solvers have also been proposed to implement the unconstrained optimization on IoT devices [15]. Optimization-based localization, however, naturally uses constraints to encode additional information, e.g. on geometry, motion or sensor topology, that are oftentimes readily available in IoT sensor deployment and can in turn improve localization accuracy. In this work we therefore propose a optimization framework for the embedded sensor self-localization that can include linear and non-linear inequality constraints with only a modest additional burden on the computation time. The implementation of the optimization framework on off-the-shelf microcontrollers further allows us to consistently compare the different optimization methods in the literature for typical indoor localization scenarios.

The self-localization approach we consider in this paper is centralized. This means that the distance measurements between all pairs of sensors is communicated to one node of the system where the self-localization is computed, and then the resulting location estimates are communicated to each sensor. In this way the communication structure required is effectively all-to-all. As sensor self-localization is usually performed on a time scale of hours to days, this communication, and it respective energy requirements, are not considered as prohibitive. As a point of contrast, the recent work of [16] presents a distributed method for self-localization of a fleet of moving agents on a time scale of minutes.

Main contributions: The main contribution of this work therefore lies in the development and the experimental implementation of an optimization-based localization framework that: i) can be implemented in a plug-and-play fashion on

IoT devices to self-localize sensors, ii) consistently compares available optimization methods for localization, iii) provides the improved localization performance in terms of accuracy and computational speed compared to the state-of-the-art in sensor self-localization, and iv) allows for additional information, specific to the IoT application at hand, to be included through inequality constraints.

Notation: Throughout the paper we use this notation:

N_{S}	number of sensors,
$\mathbf{x}_i, \hat{\mathbf{x}}_i \in \mathbb{R}^3$	respectively the true, and estimated position of sensor i in 3D space,
(x_i, y_i, z_i)	respectively the x, y, and z Cartesian coordinates of sensor position $\mathbf{x}_i$,
$d_{ij}, \tilde{d}_{ij}, \hat{d}_{ij}$	respectively the true, measured, and estimated distance between $\mathbf{x}_i$ and $\mathbf{x}_j$.

Letting $\|\cdot\|_2$ denote the standard Euclidean norm, it is clear that $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$, and $\hat{d}_{ij} = \|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j\|_2$.

II. Problem Formulation

In this work we aim to localize N_{S} sensors on a 3D field. To position all sensors in a common reference frame, we require that at least three of the sensors are not co-linear, and that a fourth sensor lies out of the plane defined by the three sensors.

A. UWB Localization Error Statistics

To obtain UWB distance measurements between the sensor locations we consider symmetric-double-sided, two-way-ranging (TWR) measurements between any pair of sensors as it ensures that clock drift is negligible [8]. This work supports the development of an experimental IoT localization system using state-of-the-art UWB radios [2]. Figure 1(a) shows the statistics of distance errors from UWB measurements for a pair of IoT devices, for eight different line-of-sight configurations. The histogram over 800 distance measurements for each configuration indicates the effect of changing relative orientation and distance on the error statistics. Hence, for the UWB devices used in the experiments, we can assume the distance measurements for each location/pair to come from a narrow distribution with a mean offset. Further, if we look at the statistics of the mean over 32 configurations, as shown in Fig. 1(b), we can conclude that the mean-offset is approximately normally distributed as distance and relative orientation changes. We have seen the same trend for all pairs of devices in our experimental implementation. These results are therefore in-line with [3], [17], namely, that we can assume the UWB distance measurements for a specific pair of devices to be corrupted with Gaussian white noise. The resulting UWB range measurements can therefore be modeled as

$$\tilde{d}_{ij} = d_{ij} + \epsilon\,, \qquad \text{with } \epsilon \sim \mathcal{N}\left(0, \sigma_{\mathrm{TWR}}\right)\,, \tag{1}$$

where σ_{TWR} denotes the standard deviation and is independent of the distance being measured. Note that, in order to obtain unbiased measurements over different locations, we have calibrated the UWB radios to sufficiently remove the effect of individual transmission delay [2].

B. Trilateration Method

Assuming that we have TWR distance measurements between all sensors, we can obtain a first estimate for all N_{S} sensors using the following trilateration steps: i) assume all measurements to be their true values, ii) use four sensors to define the datum coordinate system, and iii) estimate the position of the remaining sensors based only on distance measurements to the four datum sensors. In this way, trilateration ignores that the sensor location estimate is over-determined by the full set of distance measurements. For conciseness, we refer the reader to [1] for the trilateration equations.

C. Objective of Work

The overall objective of this work is to optimally estimate the relative position of all sensors in wireless sensor network. For the optimization-based self-localization we aim to: i) use TWR distance measurements from UWB radios installed in a 3D indoor environment, ii) implement the optimization approach on the IoT device for real-time applications, and iii) compare the performance of different non-linear optimization formulations for sensor localization.

III. Optimal Sensor Self-Localization

In this section we present different non-linear optimization formulations that have been proposed in literature for range-based localization [15]. We then provide practical guidelines for solving these problems efficiently on an embedded system, which can include encoding additional problem information such as room topology through inequality constraints.

The sensor self-localization problem can be naturally written down as an optimization problem with i) the estimated sensor locations as decision variable, ii) measured distances given as parameters of the problem, iii) the coordinate system conventions enforced with constraints, and iv) a sensible objective that is an error function between measured and estimated locations. The particular choice of objective, as introduced next, determines the accuracy and tractability of the optimization problem.

A. Minimizing distance errors

The most natural objective function is a least-squares of distance errors, the so-called *range-based least squares*, leading to the optimization problem,

$$\min_{\hat{\mathbf{x}}_i \in \mathbb{R}^3, i=1,\ldots,N_{\mathrm{S}}} \quad \sum_{i=1}^{N_{\mathrm{S}}} \sum_{i<j}^{N_{\mathrm{S}}} \left(\|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j\|_2 - \tilde{d}_{i,j} \right)^2 \tag{2a}$$

$$\text{subject to:} \quad \hat{x}_1, \hat{y}_1, \hat{z}_1, \hat{y}_2, \hat{z}_2, \hat{z}_3 \;=\; 0 \tag{2b}$$

$$\hat{x}_2, \hat{y}_3, \hat{z}_4 \;\geq\; 0\,. \tag{2c}$$

The equality constraints can be directly substituted into the objective function, resulting in an optimization problem with $3N_{\mathrm{S}} - 6$ decision variables, and three lower bound inequality constraints. Under the assumption that the d_{ij} measurements are zero-mean Gaussian distributed, as illustrated in Figure 1, this is exactly the maximum likelihood estimator of the sensor locations, and each term is weighted by the inverse of its

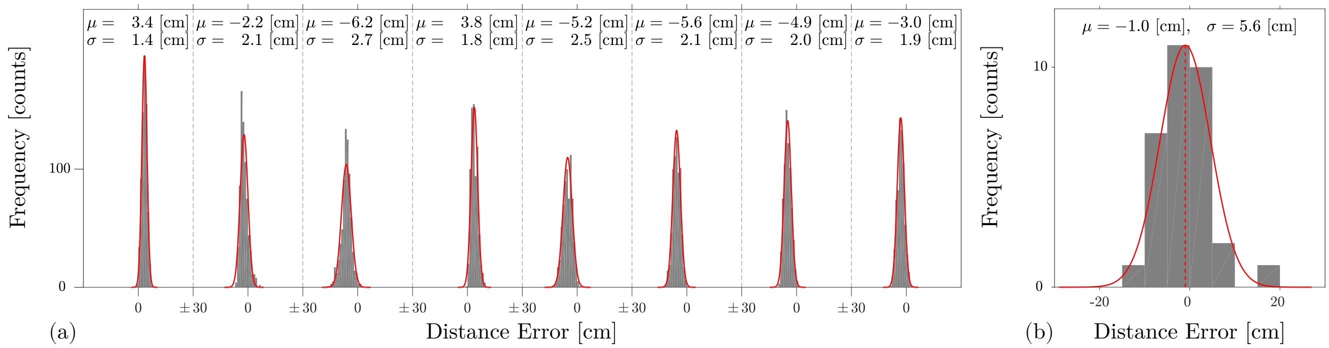

Fig. 1. (a) Experimental results showing the UWB distance errors measured between a pair of sensors for eight different locations and orientations. Each histogram (gray) shows the error statistics of 800 two-way-ranging distance measurements for a particular distance between the devices, the red line shows the normal distribution with mean, μ, and standard deviation, σ, of the data. (b) Error statistics of the mean-offset of 32 different distance and relative orientations of the same pair, i.e., the statistics of means from (a).

variance. The Euclidean norm in (2a) contains multiple terms involving the square root function which is concave and quasi-convex on $\mathbb{R}_+$, and non-differentiable at zero. The latter can occur when two sensors are co-located, i.e., if $\hat{\mathbf{x}}_i = \hat{\mathbf{x}}_j$ for $i \neq j$, which is not restrictive, as we can generally assume the true sensor locations to not be co-located.

B. Minimizing errors of squared distances

As discussed above, objective function (2a) is non-differentiable at points where two or more of the sensors are estimated to be co-located. Alternative objective functions have therefore been explored in [15], and references therein, for range-based localization. Here, we explore the *squared-range-based least squares* problem which smoothens out the non-differentiable points by minimizing the errors of the squared distances. This leads to the optimization problem,

$$\min_{\hat{\mathbf{x}}_i \in \mathbb{R}^3, i=1,\dots,N_\mathrm{S}} \quad \sum_{i=1}^{N_\mathrm{S}} \sum_{i<j}^{N_\mathrm{S}} \left(\|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j\|_2^2 - \tilde{d}_{i,j}^2 \right)^2 \tag{3a}$$

$$\text{subject to:} \quad \hat{x}_1, \hat{y}_1, \hat{z}_1, \hat{y}_2, \hat{z}_2, \hat{z}_3 = 0 \tag{3b}$$

$$\hat{x}_2, \hat{y}_3, \hat{z}_4 \geq 0\,. \tag{3c}$$

Again, the equality constraints can be substituted into the objective function, and the resulting optimization problem has the same number of decision variables and constraints as (2).

If there is no measurement noise, i.e., $\tilde{d}_{i,j} = d_{i,j}$, then the optimization problems (2) and (3) have the same optimizer, $\hat{\mathbf{x}}_i^* = \mathbf{x}_i \ \forall\, i$, and the same optimal value of zero. This is no longer true when the measurements are corrupted by noise, and the numerical results in Section IV provide empirical evidence that the solution of (2) achieves a more accurate localization.

C. Multi-dimensional scaling

Multi-dimensional scaling (MDS) is an iterative algorithm that has been researched in the literature for finding an approximate solution to problems with the form (2) and with a larger dimension [18], [19]. At each iteration, the MDS scheme minimizes a convex quadratic function that over-estimates the objective function of (2), often referred to as the *majorizing* function. The estimated locations are updated at each iteration as the minimizer of the quadratic majorizing function, and the iterations continue until the change in the objective value of (2) between subsequent iterations is less than a chosen threshold. We refer the reader to [20] for further details on the method and its convergence properties.

As efficient algorithms exist for solving quadratic programs, a potential benefit of the MDS method is computational simplicity. In addition, recent works have shown that MDS can provide high quality location estimates for sensor network self-localization, also based on UWB range measurements [14], [21]. Thus a multi-dimensional scaling method is used as the main point of comparison in Section IV where the results suggest that our proposed framework can offer improvements to both localization accuracy and computational burden.

D. Practicable IoT implementation

We have implemented the optimization problems (2) and (3) directly on an array of off-the-shelf microcontroller boards with ARM 32-bit Cortex M4 and M7 processors using the code generation software FORCES Pro [11]. We compare on these two microcontrollers because they are commonly found on IoT devices, and efficiently process float-precision and double-precision instructions respectively. Unlike other optimization software for common desktop computers, such as [22], [23], FORCES Pro generates code to solve a specific optimization problem which requires by far less memory and less computation time. All memory is statically allocated, and the generated code is library-free, which makes it easy to port it to IoT devices.

A range of iterative algorithms have been proposed in the literature for solving non-linear least-squares problems [24], each with different features and complexity. The FORCES Pro NLP solver [10] is an implementation of a line-search interior-point method similar to [23] that works with either a BFGS or GN Hessian approximation. Our numerical results in Section IV demonstrate that both variants are practical for implementation on an embedded system.

Additional information can be readily added to the optimization-based localization approach while still being a plug-and-play scheme for IoT devices and at a minimal additional computational cost. This is a major benefit of working with the natural optimization formulation and then calling on existing software to generate code for implementing iterative algorithms. Such extensions can include: i) (non-convex) inequality constraints to include information, for example, on the geometry of the surrounding environment or for sensors that lie outside of a polytope, or ii) missing distance measurements, for example, due to insufficient signal strength of non-line-of-sight communication.

For a general non-linear optimization problem, when the initial guess is sufficiently far from the optimal solution, then the algorithm may converge only to a local optimum. This issue is commonly addressed using *multi-starting* [25], where the optimization problem is solved for a set of initial conditions, and the solution with the best objective value is taken to be the optimal solution. In the Section IV results, we show empirically that multi-starting is not required, further reducing computational burden.

IV. Self-localization Results

This section presents a numerical study that demonstrates the benefits of optimization based self-localization using non-linear programming algorithms in terms of: i) localization accuracy, ii) computation times, and iii) IoT applicability, for large-scale networks with up to 30 sensors.

We consider scenarios with $N_\mathrm{S} = \{5, 6, 8, 10, 15, 20, 30\}$ sensors placed in varying configurations in a $30\mathrm{m} \times 30\mathrm{m} \times 5\mathrm{m}$ volume, each such configuration is referred to as a *field*. This scenario could represent, for example, the entrance lobby of a skyscraper. We use fields that have a horizontal expanse much greater than the vertical expanse to more adequately represent real-world indoor localization scenarios. For each instance, we generate a random field of sensors by sampling N_S locations uniformly from the volume. To make the fields more realistic, and to avoid the theoretical complications of co-located sensors, we impose that the distance between all pairs of sensors is greater than 0.5 meters. Based on the experimental results shown in Figure 1 for UWB-based TWR measurements, we generate distance measurements as per equation (1) using a standard deviation $\sigma_\mathrm{TWR} = 6$ cm.

For each set of distance measurements, we compute an estimate of the sensor locations via four methods: (i) trilateration as described in Section II-B, (ii) multi-dimensional scaling iterative scheme, referred to as *MDS (2)*, as described in Section III-C, (iii) solving optimization problem (2), referred to as *NLP (2)*, and (iv) solving (3), referred to as *NLP (3)*. For the optimization-based methods (iii) and (iv) the location estimate was obtained using the NLP solver generated by FORCES Pro [10]. To improve numerical conditioning, objective (3a) was scaled by 10^{-4}, while (2a) was left unscaled. To ensure a fair comparison we use the same initial guess for methods (ii)–(iv) and no multi-starting.

For each N_S, the four methods were used to compute a sensor localization estimate across 500 randomly generated fields. Figure 2(a) shows statistics of the localization error, represented as the euclidean distance between the true and estimated location for each sensor. The statistics are represented with a box-plot, where the box shows the inter-quartile-range, the horizontal line marks the median, and the whiskers extend to the 2^{nd} and 98^{th} percentile. Figure 2(a) shows that for five sensors all methods have similar error characteristics. This is expected as all distance measurements are used by all methods. For larger N_S, the optimization-based methods achieve approximately an order of magnitude reduction in the distance error compared to trilateration, and further improvement over the MDS method.

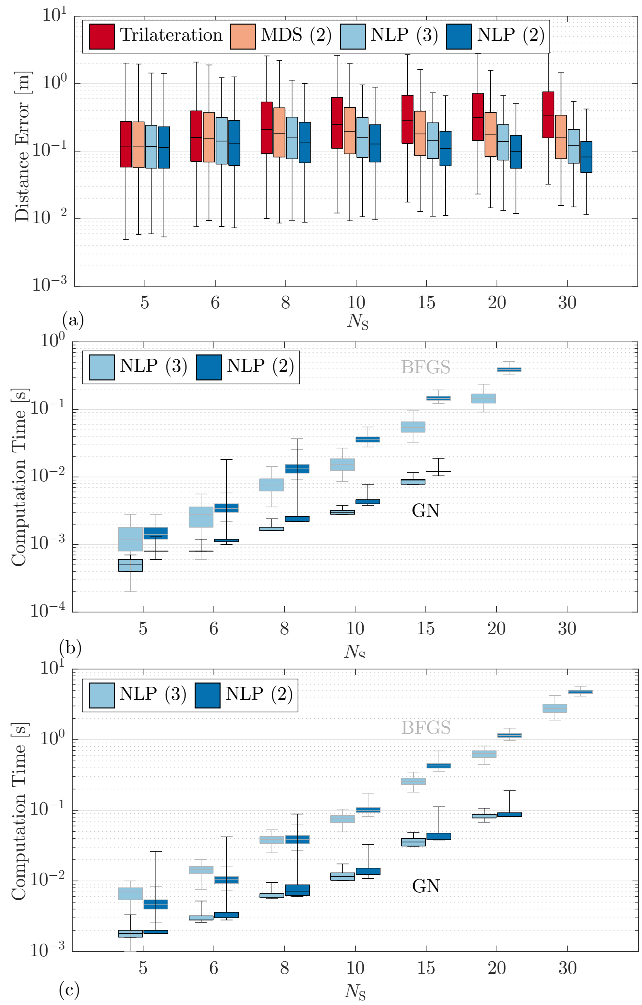

Fig. 2. Comparative study for the non-linear optimization problems (2) (dark blue shading) and (3) (light blue shading) against trilateration (red shading) and multi-dimensional scaling (light red shading), in terms of (a) the best localization accuracy represented as the statistics of distance errors for each sensor and (b) computational time required to solve the corresponding optimization problems using the BFGS (grey edges) and GN (black edges) approximations in float-precision on an ARM Cortex M4 micro-processor with 512kb of flash. (c) computational time to solve the corresponding optimization problems in double-precision on an ARM Cortex M7 micro-processor with 2Mb of flash. Where timing results are not shown the solver code was too large for the flash memory available.

The computational performance of problem (2) and problem (3) on an embedded system is pivotal to the viability of using optimization-based methods on IoT devices. Figures 2(b-c) show the statistics of the computation time required to compute a single location estimate via each method. To demonstrate the IoT applicability the FORCES Pro NLP solvers were run in: (i) float-precision on an ARM 32-bit Cortex M4 $180\mathrm{MHz}$ floating point unit (FPU) microprocessor

TABLE I
MEDIAN NUMBER OF ITERATIONS REQUIRED FOR BFGS AND GN APPROXIMATIONS TO SOLVE OPTIMIZATION PROBLEMS (2) AND (3).

N_S	Problem (2)		Problem (3)	
	BFGS	GN	BFGS	GN
5	15	4	12	4
6	22	5	20	5
8	43	5	28	6
10	67	6	32	6
15	98	7	40	6
20	125	7	49	6
30	172	7	88	7

TABLE II
COMPILED CODE SIZE IN KILOBYTES REQUIRED FOR BFGS AND GN APPROXIMATION NLP SOLVERS.

N_S	Problem (2)		Problem (3)	
	BFGS	GN	BFGS	GN
5	19	31	18	29
10	47	168	45	153
20	192	1126	182	1017
30	507	> 2000	455	> 2000

(Figure 2(b)), and (ii) double-precision on an ARM 32-bit Cortex M7 216MHz FPU microprocessor that supports double-precision instructions.

The results show that on a generic IoT device the self-localization computation time for a small sensor network ($N_S < 10$) is essentially real-time, and for a large sensor network in the order of seconds. This makes it viable to perform optimization-based sensor network self-localization directly on the IoT device, i.e. without requiring a more powerful computer to be part of the sensor network, and greatly increases the portability of a standalone IoT localization system.

Moreover, Figures 2(b-c) show that the GN method is more than an order of magnitude faster than the BFGS method. To gain some insight into the reason for this speed up, Table I provides the median number of NLP solver iterations performed to solve one problem instance. A striking feature is that for the GN method the median remains essentially constant as N_S increases. As the GN Hessian approximation is tailored to least-squares problems [26], this result suggests that it is providing a sufficient approximation of the Hessian such that the NLP algorithm mostly converges in a fixed number of iterations. This result contrasts with [8], where the authors found that the BFGS method outperformed the GN method for a similar formulation to (2). A possible explanation is that in our practical testing, we have used loose lower and upper bound constraints on all decision variables to confine them in a reasonable space, which had a positive effect on the numerical stability of the FORCES Pro NLP solver.

For the MDS method, testing indicated similar computation times for a small sensor network ($N_S < 8$), and an order of magnitude longer computation times than BFGS for large sensor networks. The main computation bottle-neck was the increasing number of iterations required for convergence as the sensor network size increased.

A drawback of the GN method is that the solver code is larger than for the BFGS method. Table II shows the compiled code size for a Cortex M7 processor. The increase in code size is mainly related to the function that computes the Hessian approximation from the Jacobian of the objective function. Table II shows that for the BFGS method the Hessian approximation requires a significantly smaller code size than for the GN method.

High reliability of the iterative NLP algorithm is also necessary for the viability of using optimization-based methods on IoT devices. To interrogate this, each instance of the NLP algorithm was categorized into the eight cases shown in Figure 3, based on convergence of the algorithm, computation time, and objective value. This breakdown suggests that GN is generally the preferred method, but more importantly it shows that for this study, solving each instance with both methods and taking the better solution allows the trilateration estimate to be improved in all instances.

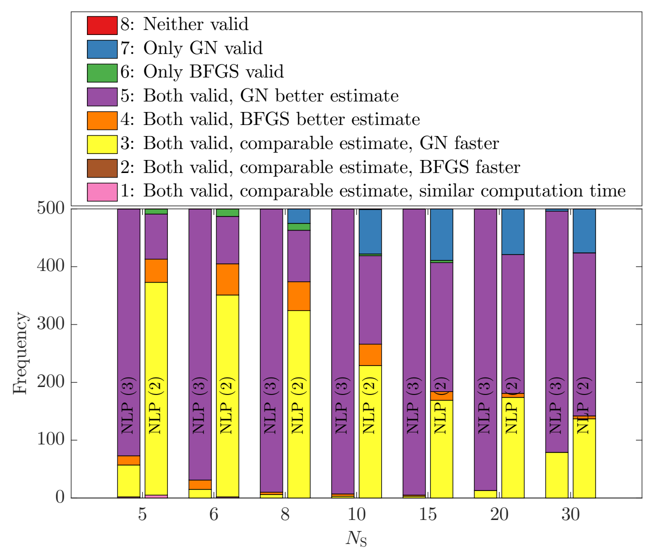

Fig. 3. Breakdown of the numerical performance of the GN and BFGS approximations to solve the non-linear optimization problems (2) and (3) in float-precision for 500 randomly generated fields.

The results presented thus far use $\sigma_{\mathrm{TWR}} = 6$ cm. To highlight that the trends are similar as σ_{TWR} is varied, Figure 4 shows the difference between the median distance error using trilateration and NLP (2) with varying N_S and σ_{TWR}. This suggests that when σ_{TWR} exceeds 1–2 cm the accuracy of using an optimization-based approach begins to justify the added computational complexity.

V. CONCLUSION

This paper proposed an optimization framework for the simultaneous self-localization of sensors in large IoT networks using range measurements from ultra-wideband radios. We have implemented two different non-linear least-squares optimization formulations tailored towards embedded self-localization. Numerical studies of a localization problem, representative of an indoor IoT scenario, demonstrate the strength of optimization-based approaches to achieve a 4-5 fold improvement in the localization accuracy when compared to trilateration which is commonly implemented in embedded

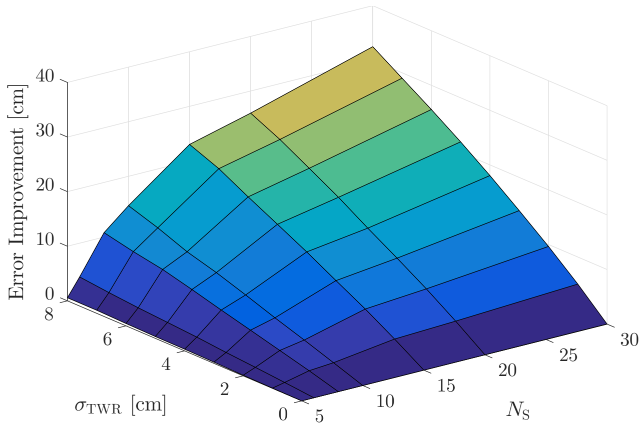

Fig. 4. Sensitivity analysis over noise levels on the distance measurements, σ_{TWR}, for all scenarios showing the error improvement over trilateration solving the non-linear optimization problem (2) using the GN approximation.

systems. We have also noticed a 2-3 fold accuracy improvement when comparing the proposed optimization formulation to the current state-of-the-art self-localization methods for wireless sensor networks.

The proposed optimization framework was implemented on off-the-shelf IoT devices in a plug-and-play manner using the software FORCES Pro to generate the iterative non-linear programming solvers required to compute locally-optimal location estimates. We analyzed two approximation techniques implemented in FORCES Pro, and consistently compared the performance with methods proposed in literature specifically for optimal sensor self-localization in terms of their numerical stability, computational cost and localization accuracy. The numerical study demonstrated that the proposed optimization framework can provide improvements not only in localization accuracy but also computational burden compared to other optimization methods proposed in literature for sensor localization. The improvements in localization accuracy of the proposed framework also stem from the ability to incorporate additional situation-specific information about the localization problem, by adding inequality constraints to the optimization problem, without affecting the viability for IoT devices.

In practical implementations, communication between pairs of devices is often blocked due to non-line-of-sight. This situation can also be easily incorporated into the framework we have presented by removing the respective term from the objective function. In future work, we will investigate how the accuracy and robustness of our optimization-based framework are influenced as the number of non-line-of-sight connections increases, and compare this with state-of-the-art methods. Moreover, to investigate how the self-localization improvements demonstrated in the numerical study transfer, we will conduct full-scale self-localization experiments for a range of sensor network arrangements. Finally, to increase portability and practicality of the proposed framework, we will adapt the code generation to compute the same Hessian approximations with a smaller code base.

References

[1] S. Gezici, Z. Tian, G. Giannakis, H. Kobayashi, A. Molisch, H. Poor, and Z. Sahinoglu, "Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 70–84, Jul 2005.

[2] *Decawave DW1000 User Manual*, Dublin, Ireland, 2016.

[3] M. Kok, J. D. Hol, and T. B. Schön, "Indoor positioning using ultra-wideband and inertial measurements," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 4, pp. 1293–1303, April 2015.

[4] K. Yu, J.-p. Montillet, A. Rabbachin, P. Cheong, and I. Oppermann, "UWB location and tracking for wireless embedded networks," *Signal Processing*, vol. 86, no. 9, pp. 2153–2171, Sep. 2006.

[5] A. Millane, T. A. Wood, H. Hesse, A. U. Zgraggen, and R. S. Smith, "Range-Inertial Estimation for Airborne Wind Energy," in *Conference on Decision and Control*, Osaka, Japan, Dec 2015, pp. 455–460.

[6] A. Chehri, P. Fortier, and P. M. Tardif, "UWB-based sensor networks for localization in mining environments," *Ad Hoc Networks*, vol. 7, no. 5, pp. 987–1000, Jul. 2009.

[7] M. W. Mueller, M. Hamer, and R. D'Andrea, "Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadrocopter state estimation," in *IEEE International Conference on Robotics and Automation*, Seattle, WA, USA, May 2015, pp. 1730–1736.

[8] K. Yu, H. Saarnisaari, J.-p. Montillet, A. Rabbachin, I. Oppermann, and G. T. Freitas de Abreu, *Localization*. Chichester, UK: John Wiley & Sons, Ltd, 2006, pp. 279–304.

[9] A. Alarifi, A. Al-Salman, M. Alsaleh, A. Alnafessah, S. Al-Hadhrami, M. Al-Ammar, and H. Al-Khalifa, "Ultra wideband indoor positioning technologies: Analysis and recent advances," *Sensors*, vol. 16, no. 5, 2016.

[10] A. Zanelli, A. Domahidi, J. Jerez, and M. Morari, "FORCES NLP: An efficient implementation of interior point methods for multistage non-linear nonconvex programs," *International Journal of Control: Special Issue on MPC Algorithms and Applications*, 2017, (to appear).

[11] A. Domahidi and J. Jerez, "FORCES Professional - embotech AG," 2014. [Online]. Available: http://embotech.com/FORCES-Pro

[12] A. H. Sayed, A. Tarighat, and N. Khajehnouri, "Network-based wireless location: challenges faced in developing techniques for accurate wireless location information," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 24–40, July 2005.

[13] Y. Wang, "Linear least squares localization in sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, p. 51, Mar 2015.

[14] C. Di Franco, A. Prorok, N. Atanasov, B. P. Kempke, P. Dutta, V. Kumar, and G. J. Pappas, "Calibration-free network localization using non-line-of-sight ultra-wideband measurements." in *IPSN*, 2017, pp. 235–246.

[15] A. Beck, P. Stoica, and J. Li, "Exact and approximate solutions of source localization problems," *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1770–1778, May 2008.

[16] I. McInerney, X. Ma, and N. Elia, "Cooperative localization from imprecise range-only measurements: A non-convex distributed approach," in *Conference on Decision and Control*, Melbourne, Australia, Dec 2017.

[17] S. Monica and G. Ferrari, "An experimental model for UWB distance measurements and its application to localization problems," in *IEEE International Conference on Ultra-WideBand*. IEEE, Sep 2014.

[18] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, no. 1, 1964.

[19] I. Borg and P. J. Groenen, *Modern multidimensional scaling: Theory and applications*. Springer Science & Business Media, 2005.

[20] J. De Leeuw, "Applications of convex analysis to multidimensional scaling," *Department of Statistics, UCLA*, 2005.

[21] J. A. Costa, N. Patwari, and A. O. Hero III, "Distributed weighted-multidimensional scaling for node localization in sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 2, no. 1, pp. 39–64, 2006.

[22] R. H. Byrd, J. Nocedal, and R. A. Waltz, "Knitro: An integrated package for nonlinear optimization," in *Large-scale nonlinear optimization*. Springer, 2006, pp. 35–59.

[23] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical programming*, vol. 106, no. 1, pp. 25–57, 2006.

[24] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer, 2006.

[25] R. Martí, *Multi-start methods*. Boston, MA: Springer, 2003.

[26] P. Chen, "Hessian matrix vs. Gauss-Newton Hessian matrix," *SIAM Journal on Numerical Analysis*, vol. 49, no. 4, pp. 1417–1435, 2011.